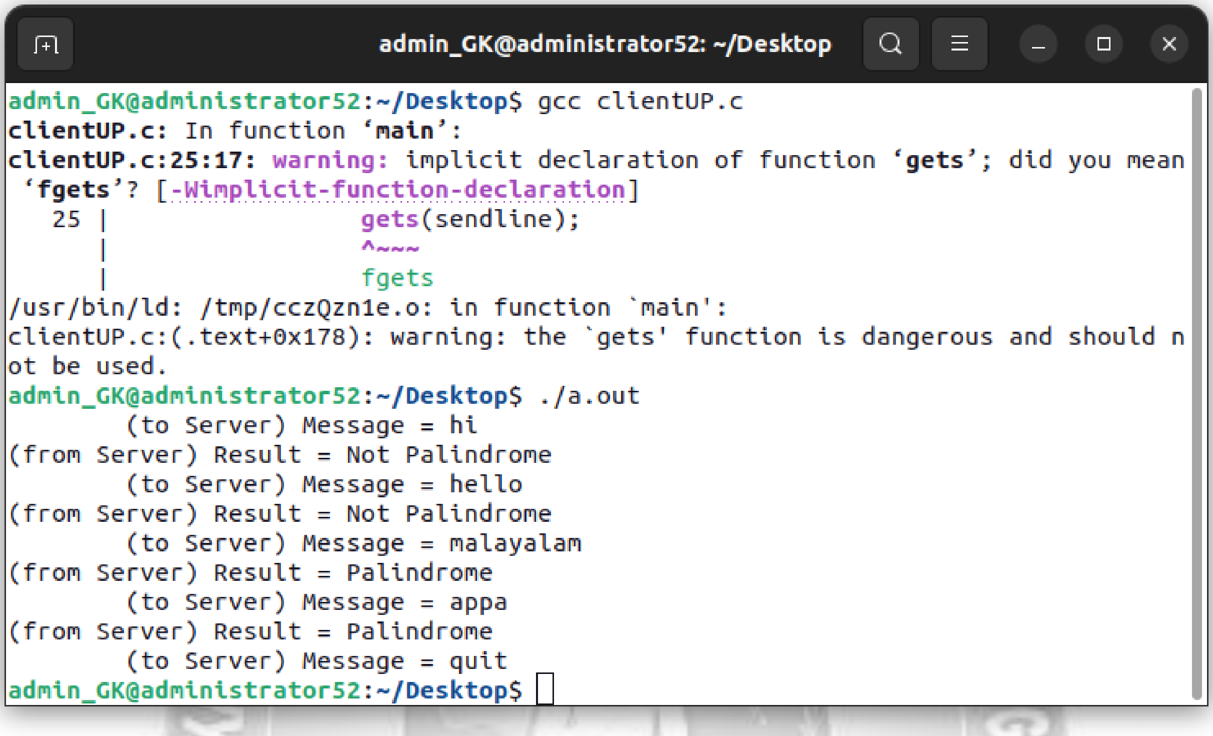
PROGRAM - serverUP.c

```
#include<stdio.h>
#include<netinet/in.h>
#include<sys/socket.h>
#include<stdlib.h>
#include<string.h>
#include<unistd.h> // close() warning removed
void main()
{
    //declaration of variables
    int s,len;
    char sendline[100]={""},recline[100]={""};
    struct sockaddr_in server,client;
    //socket creation and parameters settings
    s=socket(AF_INET,SOCK_DGRAM,0);
    server.sin_family=AF_INET;
    server.sin_port=htons(3000);
    server.sin_addr.s_addr=htonl(INADDR_ANY);
    //binding the socket
    bind(s,(struct sockaddr*)&server,sizeof(server));

    len=sizeof(client); //size of client

    while(1)
    {
        // message from client = string to check palindrome
        recvfrom(s,recline,100,0,(struct sockaddr*)&client,&len);
        if(strcmp(recline,"quit")==0)
            exit(0);
        printf("(from Client) Message = %s\n",recline);
        // message to client = palindrome or not
            //palindrome checking section
            int slen=strlen(recline);
            int end=--slen;
            int flag=0;
            for(int begin=0;begin<slen;begin++)
            {
                if(recline[begin] != recline[end])
                {
                    flag=1;
                    break;
                }
                end--;
            }
            if(flag==0)
            {
                strcpy(sendline,"Palindrome");
                sendto(s,sendline,sizeof(sendline),0,(struct sockaddr*)&client,sizeof(client));
            }
            else
            {
                strcpy(sendline,"Not Palindrome");
                sendto(s,sendline,sizeof(sendline),0,(struct sockaddr*)&client,sizeof(client));
            }
        if(strcmp(recline,"quit")==0)
            exit(0);
    }
    close(s);
}
```

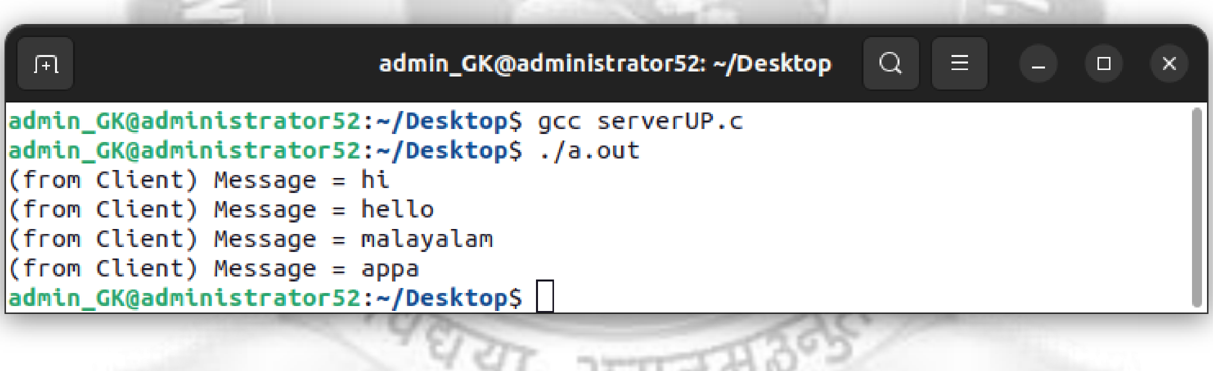
PROGRAM – clientUP.c

```
#include<stdio.h>
#include<netinet/in.h>
#include<sys/socket.h>
#include<stdlib.h>
#include<string.h>
#include<unistd.h> // close() warning removed
void main()
{
    //declaration of variables
    int c,len;
    char sendline[100]={""},recline[100]={""};
    struct sockaddr_in server,client;
    //socket creation and parameters settings
    c=socket(AF_INET,SOCK_DGRAM,0);
    server.sin_family=AF_INET;
    server.sin_port=htons(3000);
    server.sin_addr.s_addr=htonl(INADDR_ANY);

    while(1)
    {
        // message to server
        printf("    (to Server) Message = ");
        gets(sendline);
        sendto(c,sendline,sizeof(sendline),0,(struct sockaddr*)&server,sizeof(server));
        if(strcmp(sendline,"quit")==0 || strcmp(recline,"quit")==0)
            exit(0);
        // message from server
        recvfrom(c,recline,100,0,(struct sockaddr*)&client,&len);
        if(strcmp(sendline,"quit")==0 || strcmp(recline,"quit")==0)
            exit(0);
        printf("(from Server) Result = %s\n",recline);
    }
    close(c);
}
```

OUTPUT



```
admin_GK@administrator52:~/Desktop$ gcc clientUP.c
clientUP.c: In function ‘main’:
clientUP.c:25:17: warning: implicit declaration of function ‘gets’; did you mean
 ‘fgets’? [-Wimplicit-function-declaration]
   25 |                 gets(sendline);
      |                 ^~~~
      |                 fgets
/usr/bin/ld: /tmp/cczQzn1e.o: in function `main':
clientUP.c:(.text+0x178): warning: the `gets' function is dangerous and should n
ot be used.
admin_GK@administrator52:~/Desktop$ ./a.out
        (to Server) Message = hi
(from Server) Result = Not Palindrome
        (to Server) Message = hello
(from Server) Result = Not Palindrome
        (to Server) Message = malayalam
(from Server) Result = Palindrome
        (to Server) Message = appa
(from Server) Result = Palindrome
        (to Server) Message = quit
admin_GK@administrator52:~/Desktop$ 
```



```
admin_GK@administrator52:~/Desktop$ gcc serverUP.c
admin_GK@administrator52:~/Desktop$ ./a.out
(from Client) Message = hi
(from Client) Message = hello
(from Client) Message = malayalam
(from Client) Message = appa
admin_GK@administrator52:~/Desktop$ 
```